

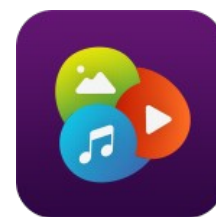
AppCritique Security Report

Qmedia

Core Report

Date Reviewed: 2019-12-01

Version Name: 1.4.2.1113



Platform	Package Name	Version Code	SHA-256 Hash
Android	com.qnap.qmediatv	12	d81c1167410bcd396c8f05e89e0a1e3 fdad8451b62d8cf50ed7a03f1612e419





Certificate Information

Name	Explanation
 Owner Name	QNAP
 Organization	QNAP
 Organizational Unit	Software Dept-II
 Location	Taipei, Taiwan, TW
 Validity	June 14, 2012 through June 08, 2037

Total number of flaws or potential flaws found: **21**

Functionality

Functionality Present

Name	Explanation
 Audio	This app plays audio and changes audio settings.
 Keychain	This app stores data in the device Keystore.
 Networking	This app connects to the internet and requests the following network related permission(s): android.permission.ACCESS_NETWORK_STATE, android.permission.ACCESS_WIFI_STATE, and android.permission.INTERNET.
 Sensors	This app uses device sensors.

Functionality Not Found

Bluetooth, Camera, Calendar, Contacts, Device Administrator, Fingerprint, Google Cloud Messaging, Geofencing, Health Data, Infrared LED, In-App Purchases, Location, Maps, Microphone, NFC, Payment Services, Photos, SMS, Telephony, USB Devices

Imported Libraries

Social Networks

Facebook	×	Flickr	×	Foursquare	×
Google+	×	Instagram	×	LinkedIn	×
Pinterest	×	Tumblr	×	Twitter	×
Yelp	×				

Analytics Networks

Adjust	×	AdobeMarketingCloud	×	AmazonAnalytics	×
AmazonInsights	×	Amplitude	×	AppBoy	×
Applause	×	Appsflyer	×	Apptimize	×
Apsalar	×	Branch	×	Countly	×
Flurry	×	GoogleAnalytics	×	Kochava	×
Localytics	×	Mixpanel	×	MobileAppTracking	×
NewRelic	×	Quantcast	×	Tapstream	×
Vessel	×	Webtrends	×		

Advertising Networks

AdColony	×	Adfalcon	×	Admob	×
AmazonAds	×	Amobee	×	AppBrain	×
AppLovin	×	Appnexus	×	Axonix	×
Chartboost	×	DoubleClick	✓	FlurryAds	×

FusePowered	×	Fyber	×	IMOB	×
Inneractive	×	Kiip	×	Liquid	×
Madvertise	×	MdotM	×	mMedia	×
Mobfox	×	MobPartner	×	NativeX	×
RevMob	×	SessionM	×	Smaato	×
Tapjoy	×				

Cloud Storage

Box	×	Cloud Drive	×	Dropbox	×
GoogleDrive	×	MediaFire	×	OneDrive	×

Developer Tools

aChartEngine	×	ActiveAndroid	×	aFileChooser	×
AmazonDeviceMessaging	×	AndroidLogger	×	Annotations	×
Answers	×	Appsee	×	AsyncHttp	×
Beacon	×	BitmapCache	×	ButterKnife	×
Digits	×	Fabric	×	Firefly	✓
Parse	×	Paypal	×	PhoneGap	×
PubNub	×	Retrofit	×	Spotify	×
Stripe	×	UniversallImageLoader	✓		

Permissions Requested

Permission Name	Is Used	Protection Level
android.permission.ACCESS_NETWORK_STATE		Normal
android.permission.ACCESS_WIFI_STATE		Normal
android.permission.FOREGROUND_SERVICE	*	
android.permission.INTERNET	✓	Normal
android.permission.MODIFY_AUDIO_SETTINGS	✓	Normal
android.permission.WAKE_LOCK	✓	Normal
com.android.providers.tv.permission.READ_EPG_DATA	*	
com.android.providers.tv.permission.WRITE_EPG_DATA	*	

* Unable to detect use of this permission.

Security Checks

Checks Conducted

Check	Result	Explanation
-------	--------	-------------

The app accesses the external storage directory, also referred to as the SDCard. External storage can be accessed by any app on a device with the READ/WRITE_EXTERNAL_STORAGE permission. It is therefore recommended not to store any sensitive information in external storage. External storage access is found in the following methods and classes:

- androidx.core.content.ContextCompat.getExternalCacheDirs(Context)
- androidx.core.content.ContextCompat.getExternalFilesDirs(Context, String)
- androidx.core.content.ContextCompat.getObbDirs(Context)
- androidx.core.content.FileProvider.parsePathStrategy(Context, String)
- androidx.core.os.EnvironmentCompat.getStorageState(File)
- com.nostra13.universalimageloader.utils.StorageUtils.getCacheDirectory(Context)
- com.nostra13.universalimageloader.utils.StorageUtils.getExternalCacheDir(Context)
- com.nostra13.universalimageloader.utils.StorageUtils.getOwnCacheDirectory(Context, String)
- com.qnap.media.SubTitleFontUtils.<init>(Context)
- com.qnap.qdk.qtshttp.mailstation.MailStation.writeStringAsFile(Context, String, String)
- com.qnap.qmediatv.AppShareData.QmediaShareResource.checkCacheSpaceAvailable(Context, QCL_AudioEntry)
- com.qnap.qmediatv.AppShareData.QmediaShareResource.getAvailableCacheDir(Context)
- com.qnap.qmediatv.MediaPlayerTv.VideoPlayer.MXPlayerSubtitleSettingFragment\$GetDownloadSubtitleUrlTask.doInBackground(String)
- com.qnap.qmediatv.MediaPlayerTv.VideoPlayer.VideoMediaPlayerGlue\$DownloadSubtitleTask.doInBackground(String)
- com.qnap.shareserverinfo.util.ServerInfoHelper.getShareServerInfoFolderPath(Context)
- com.qnap.shareserverinfo.util.ServerInfoHelper.getShareServerInfoFullPathName(Context, String)
- com.qnapcomm.common.library.boxremoteserver.QCL_BoxServerUtil
- com.qnapcomm.common.library.datastruct.QCL_FileController.copyFile(String, File)
- com.qnapcomm.common.library.datastruct.QCL_FileController.createTempDirectory()
- com.qnapcomm.common.library.datastruct.QCL_FileController.getTempDirectory()
- com.qnapcomm.common.library.datastruct.QCL_FileController.isFileExist(String)
- com.qnapcomm.common.library.datastruct.QCL_FileController.

Accesses Unique Identifiers	Not Present	The app does not access any unique identifiers.
Activities Accessible to Other Apps	Not Present	<p>No activities are exported, or access to all activities is restricted by use of permissions.</p> <p>OWASP: 2016-M1-Improper Platform Usage</p> <p>NIAP: FCS_NET_EXT.1.1</p>
Allows Data to be Backed up and Restored	Not Present	<p>The app does not allow backup of its data. This helps prevent a malicious actor with physical access to the device from dumping and analyzing user data.</p> <p>OWASP: 2016-M1-Improper Platform Usage</p>
App is Debuggable	Not Present	<p>The app is not debuggable. This protects the app against reverse engineering and the execution of arbitrary code.</p> <p>OWASP: 2016-M10-Extraneous Functionality</p>
Contains Hard-coded Cryptographic Key	Present	<p>The app contains a hard-coded cryptographic key. Cryptographic keys should never be hard-coded, as an attacker can find the key using open source and freely available tools. This may compromise the security of whatever the key is used to protect. Hard-coded cryptographic keys are found in the following classes:</p> <ul style="list-style-type: none"> •org.bouncycastle.crypto.params.DESParameters •org.bouncycastle.jce.examples.PKCS12Example •org.bouncycastle.x509.examples.AttrCertExample <p>OWASP: 2016-M5-Insufficient Cryptography; 2016-M9-Reverse Engineering</p>
Contains HostnameVerifier That Accepts All Hostnames	Present	<p>The app contains a HostnameVerifier that fails to perform hostname validation. Any network connection that uses this trust manager is vulnerable to a man-in-the-middle by an attacker using a signed certificate for a different hostname. The following methods were found to have weak hostname verifier implementations:</p> <ul style="list-style-type: none"> •com.qnap.qdk.qtshttp.QtsHttpConnection\$1.verify(String, SSLSession) •com.qnap.qdk.qtshttpapi.nassystem.Command_SSL\$1.verify(String, SSLSession) •com.qnap.qmediatv.StationWrapper.UnsafeOkHttpClient\$2.verify(String, SSLSession) <p>OWASP: 2016-M3-Insecure Communication</p> <p>NIAP: FCS_TLSC_EXT.1.2</p>

Contains Native Code	Present	<p>The app loads native code libraries. Native code does not have the same security protections as Java, and is vulnerable to buffer overflows, use after free errors, and off-by-one errors. Native code can also be loaded from untrusted sources, such as a shared directory or the network.</p> <p>OWASP: 2016-M7-Client Code Quality</p> <p>NIAP: FPT_AEX_EXT.1.5</p>
Contains Potential Hard-coded Password	Present	<p>A potential hard-coded password is found. Passwords should never be hard-coded into the app's source, as these passwords could be discovered by an attacker using open source and freely available tools. Potential passwords are found in the following classes:</p> <ul style="list-style-type: none"> •org.apache.commons.io.HexDump •org.bouncycastle.jce.examples.PKCS12Example <p>OWASP: 2016-M9-Reverse Engineering</p>
Contains Potential SQL Injection	Not Present	<p>No potential SQL injection vulnerabilities were found.</p> <p>OWASP: 2016-M7-Client Code Quality</p>
Contains Reflection Code	Present	<p>The app contains Java reflection code. Reflection is used to instantiate new objects, invoke methods, and to get and set fields at runtime. While reflection can be used for legitimate purposes, it is also commonly employed by malware to obfuscate malicious behavior.</p>

Contains X509TrustManager that Accepts All Certificates

Present

The app contains a X509TrustManager that does not validate certificates. Any network connection that uses this trust manager is vulnerable to a man-in-the-middle by an attacker using a self-signed certificate. The following methods are found to have weak trust manager implementations:

- com.qnap.qdk.qtshttpapi.nassystem.Command_SSL.checkClientTrusted(X509Certificate[], String)
- com.qnap.qdk.qtshttpapi.nassystem.Command_SSL.checkServerTrusted(X509Certificate[], String)
- com.qnap.qdk.qtshttpapi.nassystem.Command_SSL.getAcceptedIssuers()
- com.qnap.qmediatv.StationWrapper.UnsafeOkHttpClient\$1.checkClientTrusted(X509Certificate[], String)
- com.qnap.qmediatv.StationWrapper.UnsafeOkHttpClient\$1.checkServerTrusted(X509Certificate[], String)
- com.qnap.qmediatv.StationWrapper.UnsafeOkHttpClient\$1.getAcceptedIssuers()
- com.qnapcomm.common.library.util.QCL_EasyX509TrustManager.checkClientTrusted(X509Certificate[], String)
- com.qnapcomm.common.library.util.QCL_EasyX509TrustManager.checkServerTrusted(X509Certificate[], String)
- com.qnapcomm.common.library.util.QCL_EasyX509TrustManager.getAcceptedIssuers()
- com.qnapcomm.util.HttpRequestSSLUtil.getAcceptedIssuers()
- io.fabric.sdk.android.services.network.PinningTrustManager.getAcceptedIssuers()

OWASP: 2016-M3-Insecure Communication

NIAP: FIA_X509_EXT.1.1

Creates Blowfish Key with Weak Length

Not Present

The app does not create a Blowfish key with less than 128 bits in length.

OWASP: 2016-M5-Insufficient Cryptography

Creates RSA Keys with Weak Modulus Length

Not Present

The app does not create an RSA key with modulus length less than 1024 bits.

OWASP: 2016-M5-Insufficient Cryptography

NIAP: FCS_CKM.1.1(1)

Does not Update Security Provider

Not Present

The app uses the dynamic GmsCore_OpenSSL Provider to ensure that the device's security provider is always updated.

OWASP: 2016-M1-Improper Platform Usage; 2016-M5-Insufficient Cryptography

The app dynamically loads Java classes. If these classes are loaded from untrusted sources, such as a shared directory, the network, or an app from a different developer, it could be used by an attacker to gain code execution. Dynamic loading of Java code is found in the following methods:

- androidx.appcompat.app.AppCompatActivity.onCreateViewByPrefix(Context, String, String)
- androidx.appcompat.view.SupportMenuInflater\$MenuState.newInstance(String, Class, Object)
- androidx.coordinatorlayout.widget.CoordinatorLayout.parseBehavior(Context, AttributeSet, String)
- androidx.core.app.AppComponentFactory.instantiateActivityCompat(ClassLoader, String, Intent)
- androidx.core.app.AppComponentFactory.instantiateApplicationCompat(ClassLoader, String)
- androidx.core.app.AppComponentFactory.instantiateProviderCompat(ClassLoader, String)
- androidx.core.app.AppComponentFactory.instantiateReceiverCompat(ClassLoader, String, Intent)
- androidx.core.app.AppComponentFactory.instantiateServiceCompat(ClassLoader, String, Intent)
- androidx.fragment.app.Fragment.instantiate(Context, String, Bundle)
- androidx.fragment.app.Fragment.isSupportFragmentClass(Context, String)
- androidx.preference.PreferenceInflater.createItem(String, String, AttributeSet)
- androidx.recyclerview.widget.RecyclerView.createLayoutManager(Context, String, AttributeSet, int, int)
- androidx.transition.TransitionInflater.createCustom(AttributeSet, Class, String)
- com.fasterxml.jackson.databind.util.ClassUtil.getClassMethods(Class)
- edu.usf.cutr.javax.xml.stream.FactoryFinder.newInstance(String, ClassLoader)
- org.apache.commons.io.Java7Support.<clinit>()
- org.bouncycastle.jce.provider.BouncyCastleProvider.loadAlgorithms(String, String)
- org.bouncycastle.x509.X509Util.getImplementation(String, String, Provider)
- org.codehaus.stax2.validation.XMLValidationSchemaFactory.createNewInstance(ClassLoader, String)
- org.videolan.libvlc.util.HWDecoderUtil.getProperty(String, String)

OWASP: 2016-M7-Client Code Quality

NIAP: FPT_TUD_EXT.1.4

Executes Environment Commands

Present

The app executes Linux-style environment commands. This can be used to execute standard terminal commands or arbitrary files, however this requires appropriate input sanitization and/or file protections. The following method(s) utilize this:

- com.qnapcomm.debugtools.DebugLog.dumpLog(Context)
- com.qnapcomm.debugtools.LogReporter\$WriteLogThread.run()
- org.apache.commons.io.FileSystemUtils.openProcess(String)
- org.videolan.vlc.util.Logcat.getLogcat()
- org.videolan.vlc.util.Logcat.run()
- org.videolan.vlc.util.Logcat.writeLogcat(String)

OWASP: 2016-M7-Client Code Quality

The app uses a psuedo-random number generator which returns a predictable sequence of numbers that is unsuitable for security purposes. The `java.util.Random` and `java.lang.Math` classes should not be used to generate random numbers for secure use. While `SecureRandom` is the correct class for this purpose, it is recommended to avoid seeding a `SecureRandom` object. In some implementations of `SecureRandom`, seeding it may completely replace the cryptographically strong default seed. Insecure psuedo-random number generators are found in the following methods:

- `androidx.leanback.widget.StreamingTextView$DottySpan.draw(Canvas, CharSequence, int, int, float, int, int, int, Paint)`
- `androidx.transition.Explode.calculateOut(View, Rect, int)`
- `com.crashlytics.android.answers.RandomBackoff.randomJitter()`
- `com.crashlytics.android.core.CrashTest.stackOverflow()`
- `com.qnap.qdk.qtshttp.mailstation.AESHelper.getRawKey(byte)`
- `com.qnap.qdk.qtshttp.qairplay.QAirPlay.randomString(int)`
- `com.qnap.qdk.qtshttp.system.QtsHttpSystem.enableOrDisableRTRRTask(ResultEventListener, boolean, String, QtsHttpCancelController)`
- `com.qnap.qdk.qtshttp.system.QtsHttpSystem.enableOrDisableRsyncTask(ResultEventListener, boolean, int, QtsHttpCancelController)`
- `com.qnap.qdk.qtshttp.system.QtsHttpSystem.getBackupStationExtDriveList(ResultEventListener, QtsHttpCancelController)`
- `com.qnap.qdk.qtshttp.system.QtsHttpSystem.getBackupStationNasToNasList(ResultEventListener, QtsHttpCancelController)`
- `com.qnap.qdk.qtshttp.system.QtsHttpSystem.getBackupStationRTRRList(ResultEventListener, QtsHttpCancelController)`
- `com.qnap.qdk.qtshttp.system.QtsHttpSystem.getBackupStationRsyncList(ResultEventListener, QtsHttpCancelController)`
- `com.qnap.qdk.qtshttp.system.QtsHttpSystem.getMathRandom()`
- `com.qnap.qdk.qtshttp.system.QtsHttpSystem.startOrStopExtDriveTask(ResultEventListener, boolean, String, QtsHttpCancelController)`
- `com.qnap.qdk.qtshttp.system.QtsHttpSystem.startOrStopRTRRTask(ResultEventListener, boolean, String, QtsHttpCancelController)`
- `com.qnap.qdk.qtshttp.system.QtsHttpSystem.startOrStopRsyncTask(ResultEventListener, boolean, int, QtsHttpCancelController)`
- `com.qnap.qdk.qtshttpapi.musicstation.ResultController.getMathRandom()`
- `com.qnap.qdk.qtshttpapi.nassystem.ResultController.BT_Delete_Task(ResultEventListener, int, boolean, int)`
- `com.qnap.qdk.qtshttpapi.nassystem.ResultController.BT_Pause_Task(ResultEventListener, int, int)`
- `com.qnap.qdk.qtshttpapi.nassystem.ResultController.BT_Resume_Task(ResultEventListener, int, int)`

Logs Information	Present	<p>The app prints logging information to the system log. While apps often log information for debugging purposes, this should generally be removed before an app is put into production. No sensitive information, such as keys or authentication tokens, should ever be written to the system log.</p> <p>OWASP: 2016-M2-Insecure Data Storage</p> <p>NIAP: FCS_CFG_EXT.1.2</p>
Providers Accessible to Other Apps	Not Present	<p>The app does not contain content providers, no content provider is exported, or access to all content providers is restricted by use of permissions.</p> <p>OWASP: 2016-M1-Improper Platform Usage</p> <p>NIAP: FMT_CFG_EXT.1.2</p>
Receivers Accessible to Other Apps	Not Present	<p>The app does not contain receivers, no receivers are exported, or access to all exported receivers is restricted by use of permissions.</p> <p>OWASP: 2016-M1-Improper Platform Usage</p>
Requests Root Access	Not Present	<p>The app does not request root access.</p> <p>OWASP: 2016-M8-Code Tampering</p>
Services Accessible to Other Apps	Not Present	<p>The app does not contain services, no services are exported, or access to all services is restricted by use of permissions.</p> <p>OWASP: 2016-M1-Improper Platform Usage</p>
SMS CVE-2014-8610	Not Present	<p>The app does not send text messages or has the required SMS permission. It is protected from vulnerability CVE-2014-8610, where an unprivileged app can resend all the SMS stored in the user's phone to their corresponding recipients or senders without user interaction.</p> <p>OWASP: 2016-M1-Improper Platform Usage</p>
Source Code is not Obfuscated	Present	<p>The app does not obfuscate the majority of its code by renaming classes, fields, methods, and variables. This allows an adversary or competitor to decompile the app into near-original source code. It is recommended to obfuscate the app's code using a tool such as ProGuard to make it more difficult to reverse engineer.</p> <p>OWASP: 2016-M9-Reverse Engineering</p>

Uses Cipher That Does not Provide Integrity	Present	<p>The app contains a cipher does not include a HMAC (keyed-Hash Message Authentication Code) to verify the integrity of data before attempting to decrypt it. If the ciphertext can be controlled by an attacker, the cipher provides no way to detect that the data has been tampered with. Manually using an HMAC with a cipher that does not include it can lead to errors. GCM (Galois/Counter Mode) is one cipher mode that provides authenticated encryption. Ciphers using modes that do not include an HMAC are found in the following methods:</p> <ul style="list-style-type: none"> •com.qnap.qdk.qtshttp.mailstation.AESHelper.decrypt(byte[], byte[]) •com.qnap.qdk.qtshttp.mailstation.AESHelper.encrypt(byte[], byte[]) •com.qnapcomm.common.library.boxremoteserver.QCL_TasEncryptionUtil.getCipher(int) <p>OWASP: 2016-M5-Insufficient Cryptography</p>
Uses Dangerous Permissions	Not Present	<p>No dangerous permissions were requested by the app.</p> <p>OWASP: 2016-M1-Improper Platform Usage</p>
Uses DES or 3DES Cipher	Present	<p>The app uses the DES or 3DES ciphers for symmetric key encryption. DES is cryptographically insecure due to its short keylength, and can be brute forced. 3DES is subject to theoretical attacks. Generally there is no reason to use 3DES over a more modern a cipher such as AES. The following methods are found to use a DES or 3DES cipher:</p> <ul style="list-style-type: none"> •com.qnapcomm.common.library.boxremoteserver.QCL_TasEncryptionUtil.getCipher(int) <p>OWASP: 2016-M5-Insufficient Cryptography</p>
Uses Electronic Code Book Mode Cipher Mode	Present	<p>The app contains a cipher which uses the Electronic Code Book (ECB) mode. This mode does not provide good confidentiality for data because identical plaintext blocks are encrypted into identical ciphertext blocks. This can reveal patterns in the plaintext data and make protocols more susceptible to replay attacks. Ciphers using ECB mode are found in the following methods:</p> <ul style="list-style-type: none"> •com.qnap.qdk.qtshttp.mailstation.AESHelper.decrypt(byte[], byte[]) •com.qnap.qdk.qtshttp.mailstation.AESHelper.encrypt(byte[], byte[]) •com.qnapcomm.common.library.boxremoteserver.QCL_TasEncryptionUtil.getCipher(int) <p>OWASP: 2016-M5-Insufficient Cryptography</p>

Uses MD5 Hashing Algorithm	Present	<p>The app uses the weak MD5 hashing algorithm. The MD5 algorithm is dangerous if used for sensitive data because it is highly vulnerable to collision attacks.</p> <p>OWASP: 2016-M5-Insufficient Cryptography</p>
Uses NullCipher	Not Present	<p>This app does not use NullCipher.</p> <p>OWASP: 2016-M5-Insufficient Cryptography</p>

Uses Object Deserialization

Present

The app calls the `java.io.ObjectInputStream.readObject()` method to deserialize objects into memory. Object deserialization is a common source of vulnerabilities, particularly when the object may be from an untrusted source. It is recommended to avoid object deserialization when possible, or otherwise to harden the `ObjectInputStream` against attacks. One strong hardening technique is to override the `resolveClass()` to only allow expected classes. This can be implemented by subclassing `ObjectInputStream`. Object deserialization is found in the following methods:

- androidx.versionedparcelable.VersionedParcel.readSerializable()
- com.nostra13.universalimageloader.core.assist.deque.LinkedBlockingDeque.readObject(ObjectInputStream)
- com.qnap.media.QnapPlaylistPlayerFragment.load()
- com.qnap.media.QnapPlayerFragment.load()
- com.qnap.qmediatv.MediaPlayerTv.VideoPlayer.VideoMediaPlay erGlue.load()
- com.qnapcomm.base.wrapper.loginmanager.controller.QBW_Ser verController.getServerInfo(Cursor)
- org.bouncycastle.jce.provider.JCEDHPrivateKey.readObject(Obj ectInputStream)
- org.bouncycastle.jce.provider.JCEDHPublicKey.readObject(Obje ctInputStream)
- org.bouncycastle.jce.provider.JCEECPrivateKey.readObject(Obj ectInputStream)
- org.bouncycastle.jce.provider.JCEECPublicKey.readObject(Obje ctInputStream)
- org.bouncycastle.jce.provider.JCEEIGamalPrivateKey.readObjec t(ObjectInputStream)
- org.bouncycastle.jce.provider.JCEEIGamalPublicKey.readObject (ObjectInputStream)
- org.bouncycastle.jce.provider.JCERSAPrivateKey.readObject(O bjectInputStream)
- org.bouncycastle.jce.provider.JDKDSAPrivateKey.readObject(O bjectInputStream)
- org.bouncycastle.jce.provider.JDKDSAPublicKey.readObject(Obj ectInputStream)
- org.bouncycastle.jce.provider.PKCS12BagAttributeCarrierImpl.r eadObject(ObjectInputStream)

OWASP: 2016-M7-Client Code Quality

Uses RSA Encryption Algorithm
Without Padding

Not Present

The app does not use the RSA algorithm without padding.
OWASP: 2016-M5-Insufficient Cryptography

Uses SHA1 Hashing Algorithm	Present	<p>The app uses the weak SHA1 hashing algorithm. The SHA1 algorithm is dangerous if used for sensitive data because it is vulnerable to collision attacks.</p> <p>OWASP: 2016-M5-Insufficient Cryptography</p>
Weak Construction of Socket Factory	Present	<p>The app creates a SSL socket factory that may be vulnerable to man-in-the-middle (MitM) attacks. The SSLSocketFactory class creates sockets that do not automatically perform certificate hostname validation, leaving the burden on the developer to manually create and use a HostnameVerifier. Failure to do so could allow an attacker to MitM the SSL socket connection by presenting legitimate signed certificate for a different hostname. The SSLCertificateSocketFactory class creates sockets that automatically perform hostname validation, but only when instantiated with 'String host'. SSLCertificateSocketFactory also has a method 'getInsecure' that returns a SSL socket factory with all security checks disabled, which would allow an attacker to perform a MitM attack with any certificate. Google recommends choosing the highest level networking API possible, such as HttpsURLConnection, because the higher level APIs perform these security checks automatically. The following methods create SSL socket factories that either do not automatically perform hostname validation or do not automatically perform any security validation:</p> <ul style="list-style-type: none"> •com.qnapcomm.common.library.util.QCL_EasySSLSocketFactory.createSocket() •com.qnapcomm.common.library.util.QCL_EasySSLSocketFactory.createSocket(Socket, String, int, boolean) •com.squareup.okhttp.internal.http.SocketConnector.connectTls(int, int, int, Request, Route, List, boolean) <p>OWASP: 2016-M3-Insecure Communication</p>
Weak RSA Modulus Length of App Signing Certificate	Present	<p>The app is signed with a 1024 bit key, and may be vulnerable to factoring by well-funded and determined adversaries. An attacker in possession of an app's private signing key would be able to sign malicious app updates. Google recommends using a key of at least 2048 bit length.</p> <p>OWASP: 2016-M5-Insufficient Cryptography</p>

The app may use weakly configured XML parsing libraries. If XML is parsed from an untrusted source, this can lead to XML External Entity (XXE) and Denial of Service (DoS) attacks. It is recommended to use the secure processing feature to prevent DoS attacks, and disallow document type declaration (DTD) to protect against most XXE attacks. If it is not possible to disallow DTD, then disable external entities and external doctypes. Weakly configured XML parsing is found in the following methods:

- com.fasterxml.jackson.databind.ext.DOMDeserializer.parse(String)
- com.qnap.qdk.qtshttp.photostation.PhotoStation.dmcSetPlayContent(String, String[], int, QtsHttpCancelController)
- com.qnap.qdk.qtshttp.system.QtsHttpSystem.changeSystemConnectionLogStatus(ResultEventListener, QtsHttpCancelController)
- com.qnap.qdk.qtshttp.system.QtsHttpSystem.changeUserPassword(ResultEventListener, QtsHttpCancelController, String, String, String)
- com.qnap.qdk.qtshttp.system.QtsHttpSystem.createShareFolder(ResultEventListener, QtsHttpCancelController, HTTPRequestConfigDataStructure\$CreateShareFolderCTX)
- com.qnap.qdk.qtshttp.system.QtsHttpSystem.createUser(ResultEventListener, QtsHttpCancelController, HTTPRequestConfigDataStructure\$CreateUserCTX)
- com.qnap.qdk.qtshttp.system.QtsHttpSystem.createUserGroup(ResultEventListener, QtsHttpCancelController, HTTPRequestConfigDataStructure\$CreateNewUserGroupCTX)
- com.qnap.qdk.qtshttp.system.QtsHttpSystem.deleteShareFolder(ResultEventListener, QtsHttpCancelController, HTTPRequestConfigDataStructure\$DeleteShareFolderCTX)
- com.qnap.qdk.qtshttp.system.QtsHttpSystem.deleteUser(ResultEventListener, QtsHttpCancelController, String, String)
- com.qnap.qdk.qtshttp.system.QtsHttpSystem.deleteUserGroup(ResultEventListener, QtsHttpCancelController, String)
- com.qnap.qdk.qtshttp.system.QtsHttpSystem.disableAndroidStation(ResultEventListener, QtsHttpCancelController)
- com.qnap.qdk.qtshttp.system.QtsHttpSystem.disconnectExtStorageDeviceDiskPartition(ResultEventListener, QtsHttpCancelController, int)
- com.qnap.qdk.qtshttp.system.QtsHttpSystem.disconnectExtStorageDeviceDiskPartitionForSMB(ResultEventListener, QtsHttpCancelController, int)
- com.qnap.qdk.qtshttp.system.QtsHttpSystem.enableOrDisableAppCenterItem(ResultEventListener, boolean, QtsHttpCancelController, String)
- com.qnap.qdk.qtshttp.system.QtsHttpSystem.enableOrDisableDLNAQNAP(ResultEventListener, boolean, QtsHttpCancelController, HashMap)
- com.qnap.qdk.qtshttp.system.QtsHttpSystem.enableOrDisableD

Webview Contains JavaScript
Interface

**Not
Present**

The app does not expose an interface to access internal Java
methods from JavaScript.

OWASP: 2016-M7-Client Code Quality

Hard-coded Values Found

URLs	Country
http://java.sun.com/xml/stream/properties/report-cdata-event	US
http://musicbrainz.org	DE
http://ns.adobe.com/xap/1.0/	*
http://relaxng.org/ns/structure/0.9	*
http://relaxng.org/ns/structure/1.0	*
http://schemas.android.com/aapt	*
http://schemas.android.com/apk/res-auto	*
http://schemas.android.com/apk/res/android	*
http://schemas.upnp.org/upnp/1/0/	*
http://schemas.xmlsoap.org/soap/encoding/	US
http://schemas.xmlsoap.org/soap/envelope/	US
http://techslides.com/demos/sample-videos/small.mp4	US
http://www.oasis-open.org/committees/entity/release/1.0/catalog.dtd	US
Qu">http://www.qnap.com/faq/qid?lang=fr-fr>Qu	US
Cos">http://www.qnap.com/faq/qid?lang=it-it>Cos	US
http://www.satip.info/Playlists/%s.m3u	LU
http://www.shoutcast.com/sbin/newxml.phtml?genre=%s	BE
http://www.shoutcast.com/sbin/tunein-tvstation.pls?id=%s	BE
http://www.slf4j.org/codes.html#StaticLoggerBinder	CH
http://www.slf4j.org/codes.html#multiple_bindings	CH
http://www.slf4j.org/codes.html#no_static_mdc_binder	CH
http://www.slf4j.org/codes.html#null_LF	CH
http://www.slf4j.org/codes.html#null_MDCA	CH
http://www.slf4j.org/codes.html#substituteLogger	CH
http://www.slf4j.org/codes.html#unsuccessfulinit	CH
http://www.slf4j.org/codes.html#version_mismatch	CH
http://www.thaiopensource.com/trex	SG
http://www.tvdr.de/	DE
http://www.videolan.org	FR
http://www.videolan.org/vlc/playlist/0	FR
http://www.w3.org/1999/xhtml	US

http://www.w3.org/2000/xmlns	US
http://www.w3.org/2000/xmlns/	US
http://www.w3.org/2000/xmlns/	US
http://www.w3.org/2001/XInclude	US
http://www.w3.org/2001/XMLSchema	US
http://www.w3.org/2001/XMLSchema-datatypes	US
http://www.w3.org/2001/XMLSchema-instance	US
http://www.w3.org/2002/08/xquery-functions	US
http://www.w3.org/2003/XInclude	US
http://www.w3.org/2004/11/ttaf1	US
http://www.w3.org/2006/04/ttaf1	US
http://www.w3.org/2006/10/ttaf1	US
http://www.w3.org/TR/REC-html40/loose.dtd	US
http://www.w3.org/TR/html4/strict.dtd	US
http://www.w3.org/TR/xhtml1/DTD/xhtml1-frameset.dtd	US
http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd	US
http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd	US
http://www.w3.org/TR/xhtml10/DTD/xhtml10strict.dtd	US
http://www.w3.org/XML/1998/namespace	US
http://www.w3.org/XML/1998/namespace)	US
http://www.w3.org/XML/1998/namespace\	US
http://www.w3.org/ns/ttml	US
http://xmlpull.org/v1/doc/features.html#names-interned	US
http://xmlpull.org/v1/doc/features.html#process-docdecl	US
http://xmlpull.org/v1/doc/features.html#process-namespaces	US
http://xmlpull.org/v1/doc/features.html#xml-roundtrip	US
https://account.qnap.com.cn/mobilesignin?state=%2Foauth%2Fauth%3Fresponse_type%3Dtoken	CN
https://account.qnap.com.cn/v2/mobilesignin?state=%2Foauth%2Fauth%3Fresponse_type%3Dtoken	CN
https://account.qnap.com/mobilesignin?state=%2Foauth%2Fauth%3Fresponse_type%3Dtoken	US
https://account.qnap.com/v2/mobilesignin?state=%2Foauth%2Fauth%3Fresponse_type%3Dtoken	US
https://alpha-account.qnap.com.cn/mobilesignin?state=%2Foauth%2Fauth%3Fresponse_type%3Dtoken	CN
https://alpha-account.qnap.com.cn/v2/mobilesignin?state=%2Foauth%2Fauth%3Fresponse_type%3Dtoken	CN
https://alpha-account.qnap.com/mobilesignin?state=%2Foauth%2Fauth%3Fresponse_type%3Dtoken	US

https://alpha-account.qnap.com/v2/mobilesignin?state=%2Foauth%2Fauth%3Fresponse_type%3Dtoken	US
https://e.crashlytics.com/spi/v2/events	US
https://fingerprint.videolan.org/acoustid.php?meta=recordings+tracks+usermeta+releases&duration=%d&fingerprint=%s	FR
https://pagead2.googlesyndication.com/pagead/gen_204?id=gmob-apps	US
https://play.google.com/store/apps/details?id=	US
https://play.google.com/store/apps/details?id=com.google.android.apps.authenticator2	US
https://settings.crashlytics.com/spi/v2/platforms/android/apps/%s/settings	US
https://update.qnap.com.cn/SoftwareReleaseS.xml	CN
https://update.qnap.com/SoftwareReleaseS.xml	US
https://videos.pexels.com/videos/birds-on-street-at-city-485	US
https://videos.pexels.com/videos/greenhouse-with-cacti-534	US
https://videos.pexels.com/videos/roller-coaster-560	US
https://videos.pexels.com/videos/timelapse-of-stockholm-at-night-539	US
https://videos.pexels.com/videos/waterfall-stream-nature-423	US
https://www.googleapis.com/auth/appstate	US
https://www.googleapis.com/auth/datastoremobile	US
https://www.googleapis.com/auth/drive.appdata	US
https://www.googleapis.com/auth/drive.file	US
https://www.googleapis.com/auth/fitness.activity.read	US
https://www.googleapis.com/auth/fitness.activity.write	US
https://www.googleapis.com/auth/fitness.body.read	US
https://www.googleapis.com/auth/fitness.body.write	US
https://www.googleapis.com/auth/fitness.location.read	US
https://www.googleapis.com/auth/fitness.location.write	US
https://www.googleapis.com/auth/fitness.nutrition.read	US
https://www.googleapis.com/auth/fitness.nutrition.write	US
https://www.googleapis.com/auth/games	US
https://www.googleapis.com/auth/games_lite	US
https://www.googleapis.com/auth/plus.login	US
https://www.googleapis.com/auth/plus.me	US
https://www.qnap.com/about?lang=	US
https://www.qnap.com/consent-to-use-of-data?lang=	US
https://www.qnap.com/go/how-to/faq/article/why-do-qnap-mobile-apps-and-utilities-need-to-confirm-	US

my-current-location?lang=

telnet://0.0.0.0:4212

*

IP Addresses

Country

1.3.132.0

*

104.199.156.58

*

120.24.59.150

*

127.0.0.1

*

172.16.0.0

*

192.168.0.0

*

192.168.0.1

*

192.168.1.1

*

198.16.70.58

*

2.23.136.1

*

2.5.24.72

*

2.5.29.14

*

2.5.29.15

*

2.5.29.16

*

2.5.29.17

*

2.5.29.18

*

2.5.29.19

*

2.5.29.20

*

2.5.29.21

*

2.5.29.23

*

2.5.29.24

*

2.5.29.27

*

2.5.29.28

*

2.5.29.29

*

2.5.29.30

*

2.5.29.31

*

2.5.29.32

*

2.5.29.33

*

2.5.29.35

*

2.5.29.36

*

2.5.29.37	*
2.5.29.46	*
2.5.29.54	*
2.5.29.55	*
2.5.29.56	*
255.255.255.255	*
45.79.174.251	*
50.19.254.134	*

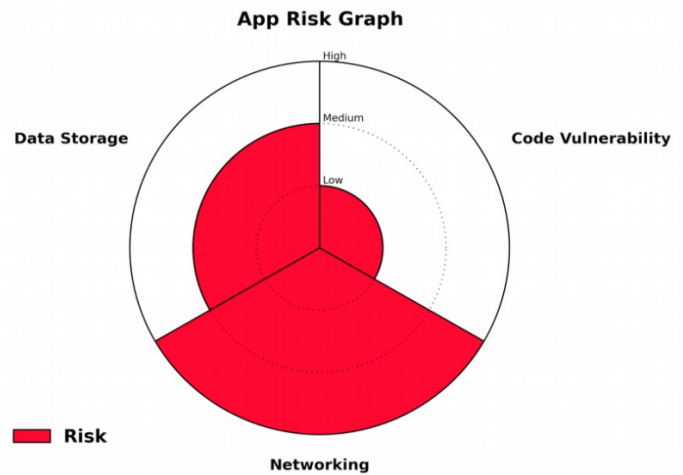
Emails

appro@openssl.org
 feedback-crypto@bouncycastle.org
 ffmpeg-devel@ffmpeg.org
 fstatvfs@openssh.com
 fsync@openssh.com
 hmac-ripemd160@openssh.com
 keepalive@libssh2.org
 libdvbpsi-devel@videolan.org
 mobile@qnap.com
 rijndael-cbc@lysator.liu.se
 sam@zoy.org
 statvfs@openssh.com
 xxx@qnap.com
 zlib@openssh.com

Upgrade to App Vulnerability Assessment

Thank you for trying the free AppCritique scan! Data breaches lead to loss of intellectual property, litigation, customer dissatisfaction and loss in revenue. Many significant vulnerabilities cannot be detected by our Free Report. AppCritique offers the **App Vulnerability Assessment (AVA)** as a deep dive of your app conducted by the AppCritique experts. The AVA service includes:

- Expert analysis of your app with risk assessment write-ups
- Recommendations and remediations
- Q&A session between your app developers or assurance team and the expert AppCritique analysts
- AVA checks include:
 - **Code Vulnerabilities**
 - Inter-app communications
 - Components vulnerable to manipulation
 - Unauthenticated or unfiltered input
 - Local SQL injection
 - Unsafe native code
 - Dynamically loaded code
 - Hard-coded credentials
 - Deprecated cryptography
 - **Data Storage**
 - Data accessible in unencrypted backups
 - Publicly accessible sensitive information
 - Credentials outside secure store
 - Data privacy analysis
 - Side channel data leakage
 - **Networking**
 - Certificate validation issues
 - Unencrypted protocols
 - Weak endpoint encryption
 - Back-end analysis
 - Man-in-the-middle attack scenarios



Sample App Risk Graph

Contact Us

Email us at AppCritique@bah.com to set up your App Vulnerability Assessment!